

Remarks

Reconsideration of the application is respectfully requested.

Claims 1-21 were rejected in the Office Action mailed April 18, 2006. Of those claims, claims 11, 16, and 21 have been amended to correct informalities. Accordingly, claims 1-21 remain pending in the application.

Claim Rejections under 35 U.S.C. § 102

In "Claim Rejections – 35 USC § 102" item 2 on page 2 of the above-identified Office Action, claims 1-2, 5-6, 11-12, 15-16, and 21 have been rejected as being fully anticipated by U.S. Patent No. 6,675,353 to *Friedman* (hereinafter "Friedman") under 35 U.S.C. § 102(e). Applicants respectfully traverse.

Claim 1 recites a method of computing comprising:
"reading and parsing a data processing representation;
recognizing a declaration reference to an executable namespace;
recognizing an expression referencing a function of the executable
namespace;

instantiating the referenced function or a function creator to create the
function, then instantiate the created function; and
evaluating the expression using the instantiated function."

In contrast, Friedman simply teaches a method of generating an XML document that does not disclose, expressly or inherently, recognizing expressions referencing functions of namespaces, instantiating those functions (including creating the functions, if necessary), and evaluating the expressions using the functions, as is claimed by claim 1. Rather, Friedman teaches a request object that may be implemented as a C++ class which receives request information from a client application and, in response, generates an XML document. The client application may transmit namespace values and property values associated with the namespaces through API functions of the request object, such as "AddNamespace"

and “AddProperty.” The request object may receive the values, may store the values in a data structure, and may assign a moniker to a namespace value to return to the client application. The moniker may then be used by the client application to associate the property values it transmits with the appropriate namespaces. Upon receiving all of the information from the client application, the request object may invoke one of its methods to generate an XML document.

Friedman’s “request” is NOT a “data processing representation” as the term “data processing representation” is used and defined by Applicants in the Specification. Accordingly, Friedman does not anticipate the recited “parsing a data processing representation”.

Even if we are to ignore this deficiency and assume, for the sake of argument, that Friedman’s processing of the “request” may be properly read as anticipating “parsing a data processing representation”, it is NOT inherent that, for Friedman’s client application to provide namespace and property values for the generation of the XML document, the request MUST include “a declaration referencing an executable namespace” and “an expression referencing a function of the executable namespace”, which the “generator” of the XML document MUST recognize. Friedman’s API may be implemented in such a manner that the namespace may be referenced without employing a “declaration”, for example, merely as a parameter of the Add Namespace function, which is NOT a “declaration”, in accordance with the term’s plain meaning, as the term is understood by those of ordinary skill in the art. Likewise, Friedman’s provision of namespace and property values may be implemented without ever requiring the processing logic of the “request” to recognize that a function of the namespace is being referenced through an “expression”, as the term “expression” is commonly understood by those of ordinary skill in the art.

The functions taught by Friedman, such as “AddNamespace”, “AddProperty”, and the methods of the request object are NOT functions of the referenced namespace, and therefore simply do not read on “a function of the executable namespace.” As the term is used in the Specification of the present application, a “namespace” is identified by a URI representing a hostname or path. A “function of the executable namespace” is a function located on the host or in the path

represented by the URI identifying the namespace (see pgs. 5-6 of the Specification of the present application, discussing the following exemplary expression referencing a function of an executable namespace: "<math:add xmlns:math='x://bestuniversity.edu/mathdept/mathlibr/'>"). In the parenthetical example, "math:add" is the "expression referencing a function" (here, a function named "add"), where "add" is a function that may be found at the URI "bestuniversity.edu/mathdept/mathlibr/." None of the functions disclosed by Friedman are taught as being located on a host or in a path identified by the namespace value.

Even if we ignore all the deficiencies discussed thus far, Friedman still fails to teach "instantiating the referenced function", including creating the referenced function if necessary, and finally "evaluating the expression using the instantiated function." At best, Friedman generates the XML document with the referenced functions of the referenced namespace, for some other processing logic adapted to process the generated XML document to instantiate the referenced function of the referenced namespace, and to evaluate the recognized expression referencing the function.

Accordingly, claim 1 is not anticipated by Friedman under 35 USC §102(e).

Claims 11 and 21 recite similar limitations similar to those of claim 1. Thus, for at least the same reasons, claims 11 and 21 are not anticipated by Friedman.

Claims 2, 5-6, 12, and 15-16 depend from claims 1 and 11, incorporating their limitations respectively. Thus, for at least the same reasons, claims 2, 5-6, 12, and 15-16 are not anticipated by Friedman.

Claim Rejections under 35 U.S.C. § 103

In "Claim Rejections – 35 USC § 103" item 4 on page 4 of the above-identified Office Action, claims 3-4, 7-10, 13-14, and 17-20 have been rejected as

being unpatentable over Friedman under 35 U.S.C. § 103. Applicants respectfully traverse.

Claims 3-4, 7-10, 13-14, and 17-20 depend from claims 1 and 11, incorporating their limitations respectively. Thus, for at least the same reasons that claims 1 and 11 are not anticipated by or obvious in view of Friedman, claims 3-4, 7-10, 13-14, and 17-20 are patentable over Friedman.

Conclusion

In view of the foregoing, reconsideration and allowance of claims 1-21 are solicited. As a result of the amendments made herein, Applicant submits that claims 1-21 are in condition for allowance. Accordingly, a Notice of Allowance is respectfully requested. If the Examiner has any questions concerning the present paper, the Examiner is kindly requested to contact the undersigned at (206) 407-1513. If any fees are due in connection with filing this paper, the Commissioner is authorized to charge the Deposit Account of Schwabe, Williamson and Wyatt, P.C., No. 50-0393.

Respectfully submitted,
SCHWABE, WILLIAMSON & WYATT, P.C.

Date: May 24, 2006

by: 
Robert C. Peck
Reg. No.: 56,826

Schwabe, Williamson & Wyatt, P.C.
Pacwest Center, Suites 1600-1900
1211 SW Fifth Avenue
Portland, Oregon 97222
Telephone: 503-222-9981